

30+ Basic HTML Coding Projects for Class 10

Introduction

Welcome to this exciting journey of learning HTML! This guide is packed with over 30+ hands-on projects that will help you understand the fundamentals of web development. By the end, you'll be able to create your own websites using HTML. Each project includes explanations and source code to make learning easy and fun!

Visit: [Modern Age Coders](#) to learn more about coding

Chapter 1: Getting Started with HTML

Project 1: Your First Webpage

Objective:

Learn the structure of an HTML document and create a simple webpage.

Explanation:

Every webpage starts with a basic structure that includes a `<!DOCTYPE html>` declaration, an `<html>` tag, a `<head>` section, and a `<body>` section.

Source Code:

```
<!DOCTYPE html>
<html>
<head>
  <title>My First Webpage</title>
</head>
<body>
  <h1>Welcome to My First Webpage</h1>
  <p>This is my first time coding in HTML!</p>
</body>
</html>
```

Chapter 2: Working with Text and Formatting

Project 2: Adding Headings and Paragraphs

Objective:

Understand how to format text using headings and paragraphs.

Explanation:

Headings range from `<h1>` to `<h6>` and are used to structure content. Paragraphs are enclosed in `<p>` tags.

Source Code:

```
<!DOCTYPE html>
<html>
<head>
  <title>Headings and Paragraphs</title>
</head>
<body>
  <h1>Main Heading</h1>
  <h2>Subheading</h2>
  <p>This is a paragraph explaining the topic in detail.</p>
</body>
</html>
```

Chapter 3: Lists and Links

Project 3: Creating an Ordered and Unordered List

Objective:

Learn to organize content using ordered (``) and unordered (``) lists.

Source Code:

```
<!DOCTYPE html>
```

```
<html>
<head>
  <title>Lists in HTML</title>
</head>
<body>
  <h1>My Favorite Fruits</h1>
  <ul>
    <li>Apple</li>
    <li>Banana</li>
    <li>Orange</li>
  </ul>
  <h2>Steps to Make a Sandwich</h2>
  <ol>
    <li>Get two slices of bread</li>
    <li>Spread butter on one slice</li>
    <li>Add fillings of your choice</li>
    <li>Cover with the other slice</li>
  </ol>
</body>
</html>
```

Chapter 4: Adding Images and Links

Project 4: Inserting Images

Objective:

Learn how to add images using the `` tag.

Source Code:

```
<!DOCTYPE html>
<html>
<head>
  <title>Adding Images</title>
</head>
<body>
  <h1>My Favorite Animal</h1>
  
</body>
</html>
```

Project 5: Creating a Navigation Menu

Objective:

Use the `<a>` tag to create links for navigation.

Source Code:

```
<!DOCTYPE html>
<html>
<head>
  <title>Navigation Menu</title>
</head>
<body>
  <h1>My Website</h1>
  <nav>
    <a href="home.html">Home</a> |
    <a href="about.html">About</a> |
    <a href="contact.html">Contact</a>
  </nav>
</body>
</html>
```

Chapter 5: Forms and Input Fields

Project 6: Creating a Simple Form

Objective:

Learn how to create a form with input fields.

Source Code:

```
<!DOCTYPE html>
<html>
<head>
  <title>Navigation Menu</title>
</head>
<body>
```

```
<h1>My Website</h1>
<nav>
  <a href="home.html">Home</a> |
  <a href="about.html">About</a> |
  <a href="contact.html">Contact</a>
</nav>
</body>
</html>
```

Project 12: Building a Personal Portfolio Website

Objective:

Learn how to create a personal portfolio page with your work samples.

Explanation:

A portfolio page includes sections like an introduction, work samples, and contact information.

Source Code:

```
html
CopyEdit
<!DOCTYPE html>
<html>
<head>
  <title>My Portfolio</title>
</head>
<body>
  <h1>Welcome to My Portfolio</h1>
  <section>
    <h2>About Me</h2>
    <p>I am a passionate web developer!</p>
  </section>
  <section>
    <h2>My Projects</h2>
    <ul>
      <li>Project 1: Website Design</li>
      <li>Project 2: E-commerce Site</li>
    </ul>
  </section>
```

```
<section>
  <h2>Contact Me</h2>
  <p>Email: example@example.com</p>
</section>
</body>
</html>
```

Project 13: Creating a Table Layout for a Blog

Objective:

Learn how to use tables to design a simple blog layout.

Explanation:

This layout will include a navigation bar, main content area, and sidebar using a table structure.

Source Code:

html

CopyEdit

```
<!DOCTYPE html>
<html>
<head>
  <title>Blog Layout</title>
</head>
<body>
  <table border="1">
    <tr>
      <td colspan="2" style="text-align:center;">My Blog</td>
    </tr>
    <tr>
      <td width="75%">
        <h2>Blog Post</h2>
        <p>This is where the blog post content will go.</p>
      </td>
      <td width="25%" style="background-color: lightgray;">
        <h3>Sidebar</h3>
        <ul>
          <li><a href="#">About Me</a></li>
```

```
        <li><a href="#">Contact</a></li>
    </ul>
</td>
</tr>
</table>
</body>
</html>
```

Project 14: Contact Form with Validation

Objective:

Create a contact form with input validation.

Explanation:

Learn to use the `required` attribute to ensure users fill out necessary fields before submitting.

Source Code:

```
html
CopyEdit
<!DOCTYPE html>
<html>
<head>
    <title>Contact Form</title>
</head>
<body>
    <h1>Contact Us</h1>
    <form action="/submit" method="POST">
        <label for="name">Name:</label><br>
        <input type="text" id="name" name="name" required><br><br>

        <label for="email">Email:</label><br>
        <input type="email" id="email" name="email" required><br><br>

        <input type="submit" value="Submit">
    </form>
</body>
</html>
```

Project 15: Responsive Web Design with Media Queries

Objective:

Learn how to make your website responsive using CSS media queries.

Explanation:

Responsive design ensures your website looks good on all devices by adjusting layouts based on screen size.

Source Code:

html

CopyEdit

```
<!DOCTYPE html>
<html>
<head>
  <title>Responsive Layout</title>
  <style>
    body {
      font-family: Arial, sans-serif;
    }
    .content {
      width: 80%;
      margin: 0 auto;
    }
    @media (max-width: 600px) {
      .content {
        width: 100%;
      }
    }
  </style>
</head>
<body>
  <div class="content">
    <h1>Welcome to My Responsive Website</h1>
    <p>This layout adjusts based on screen size!</p>
  </div>
```



```
</body>
</html>
```

I can continue expanding with more projects as needed! Would you like me to proceed with additional examples or another specific section?

You said:

yes cover remaining 25 projects!

ChatGPT said:

Sure! Let's continue with the remaining projects. Here are the next 25 projects to complete the full set of 40+ HTML projects.

Project 16: Simple Image Gallery

Objective:

Create an image gallery to display pictures in a grid layout.

Explanation:

You'll use `` tags within a grid layout to display images.

Source Code:

html

CopyEdit

```
<!DOCTYPE html>
<html>
<head>
  <title>Image Gallery</title>
  <style>
    .gallery {
      display: grid;
      grid-template-columns: repeat(3, 1fr);
      gap: 10px;
    }
    .gallery img {
      width: 100%;
      height: auto;
    }
  </style>
</head>
<body>
  <div class="gallery">
    <img alt="Placeholder for image 1" data-bbox="111 911 195 927" />
    <img alt="Placeholder for image 2" data-bbox="206 911 290 927" />
    <img alt="Placeholder for image 3" data-bbox="301 911 385 927" />
    <img alt="Placeholder for image 4" data-bbox="396 911 480 927" />
    <img alt="Placeholder for image 5" data-bbox="491 911 575 927" />
    <img alt="Placeholder for image 6" data-bbox="586 911 670 927" />
    <img alt="Placeholder for image 7" data-bbox="681 911 765 927" />
    <img alt="Placeholder for image 8" data-bbox="776 911 860 927" />
    <img alt="Placeholder for image 9" data-bbox="871 911 955 927" />
    <img alt="Placeholder for image 10" data-bbox="966 911 1050 927" />
    <img alt="Placeholder for image 11" data-bbox="1061 911 1145 927" />
    <img alt="Placeholder for image 12" data-bbox="1156 911 1240 927" />
    <img alt="Placeholder for image 13" data-bbox="1251 911 1335 927" />
    <img alt="Placeholder for image 14" data-bbox="1346 911 1430 927" />
    <img alt="Placeholder for image 15" data-bbox="1441 911 1525 927" />
    <img alt="Placeholder for image 16" data-bbox="1536 911 1620 927" />
    <img alt="Placeholder for image 17" data-bbox="1631 911 1715 927" />
    <img alt="Placeholder for image 18" data-bbox="1726 911 1810 927" />
    <img alt="Placeholder for image 19" data-bbox="1821 911 1905 927" />
    <img alt="Placeholder for image 20" data-bbox="1916 911 2000 927" />
    <img alt="Placeholder for image 21" data-bbox="2011 911 2095 927" />
    <img alt="Placeholder for image 22" data-bbox="2106 911 2190 927" />
    <img alt="Placeholder for image 23" data-bbox="2201 911 2285 927" />
    <img alt="Placeholder for image 24" data-bbox="2296 911 2380 927" />
    <img alt="Placeholder for image 25" data-bbox="2391 911 2475 927" />
  </div>
</body>
</html>
```

```
        }
    </style>
</head>
<body>
    <h1>My Image Gallery</h1>
    <div class="gallery">
        
        
        
    </div>
</body>
</html>
```

Project 17: Simple Calculator

Objective:

Create a simple calculator using HTML and JavaScript.

Explanation:

This project combines HTML and JavaScript to create interactive buttons for the calculator.

Source Code:

```
html
CopyEdit
<!DOCTYPE html>
<html>
<head>
    <title>Calculator</title>
</head>
<body>
    <h1>Simple Calculator</h1>
    <form name="calculator">
        <input type="text" name="display" id="display">
        <br>
        <button type="button" onclick="addToDisplay('1')">1</button>
        <button type="button" onclick="addToDisplay('2')">2</button>
        <button type="button" onclick="addToDisplay('3')">3</button>
```

```
        <br>
        <button type="button" onclick="calculate()">=</button>
        <button type="button" onclick="clearDisplay()">C</button>
    </form>

    <script>
        function addToDisplay(value) {
            document.calculator.display.value += value;
        }

        function calculate() {
            document.calculator.display.value =
eval(document.calculator.display.value);
        }

        function clearDisplay() {
            document.calculator.display.value = '';
        }
    </script>
</body>
</html>
```

Project 18: Animated Text with CSS

Objective:

Animate text using CSS keyframes.

Explanation:

CSS animations allow you to create smooth transitions and movements on your webpage.

Source Code:

```
html
CopyEdit
<!DOCTYPE html>
<html>
<head>
    <title>Animated Text</title>
```

```
<style>
  @keyframes fadeIn {
    0% { opacity: 0; }
    100% { opacity: 1; }
  }
  h1 {
    animation: fadeIn 2s ease-in-out;
  }
</style>
</head>
<body>
  <h1>Welcome to My Website</h1>
</body>
</html>
```

Project 19: Simple To-Do List

Objective:

Create a simple to-do list where users can add and remove items.

Explanation:

You'll use a combination of HTML and JavaScript to add interactivity.

Source Code:

```
html
CopyEdit
<!DOCTYPE html>
<html>
<head>
  <title>To-Do List</title>
</head>
<body>
  <h1>My To-Do List</h1>
  <ul id="todo-list">
    <li>Buy groceries</li>
    <li>Complete homework</li>
  </ul>
```

```
<input type="text" id="new-task" placeholder="Add a new task">
<button onclick="addTask()">Add Task</button>

<script>
    function addTask() {
        var task = document.getElementById("new-task").value;
        if (task) {
            var li = document.createElement("li");
            li.textContent = task;
            document.getElementById("todo-list").appendChild(li);
            document.getElementById("new-task").value = "";
        }
    }
</script>
</body>
</html>
```

Project 20: Personal Blog Layout

Objective:

Create a simple layout for a personal blog with articles and sidebar.

Explanation:

This layout will use a combination of `<section>`, `<article>`, and a sidebar.

Source Code:

```
html
CopyEdit
<!DOCTYPE html>
<html>
<head>
    <title>Personal Blog</title>
    <style>
        body {
            font-family: Arial, sans-serif;
        }
        .content {
```

```
        display: flex;
    }
    .main-content {
        width: 75%;
    }
    .sidebar {
        width: 25%;
        background-color: lightgray;
    }
</style>
</head>
<body>
    <h1>Welcome to My Blog</h1>
    <div class="content">
        <div class="main-content">
            <h2>Blog Post Title</h2>
            <p>This is the content of the blog post.</p>
        </div>
        <div class="sidebar">
            <h3>About Me</h3>
            <p>I'm a web developer!</p>
        </div>
    </div>
</body>
</html>
```

Project 21: Interactive Image Map

Objective:

Create an interactive image map with clickable regions.

Explanation:

Use the `<map>` and `<area>` tags to define clickable areas on an image.

Source Code:

html

CopyEdit

```
<!DOCTYPE html>
<html>
<head>
  <title>Image Map</title>
</head>
<body>
  <h1>Interactive Image Map</h1>
  
  <map name="worldmap">
    <area shape="rect" coords="34,44,270,350" alt="USA"
href="usa.html">
    <area shape="circle" coords="300,250,50" alt="Europe"
href="europe.html">
  </map>
</body>
</html>
```

Project 22: Weather Information Page

Objective:

Create a weather page with basic information and images.

Explanation:

This page will display weather-related data, like temperature and condition.

Source Code:

```
html
CopyEdit
<!DOCTYPE html>
<html>
<head>
  <title>Weather Information</title>
</head>
<body>
  <h1>Weather Forecast</h1>
  <p>Temperature: 25°C</p>
  <p>Condition: Sunny</p>
```

```
        
</body>
</html>
```

Project 23: Creating a Footer Section

Objective:

Learn how to design a footer for a website.

Explanation:

The footer typically includes contact information or copyright details.

Source Code:

```
html
CopyEdit
<!DOCTYPE html>
<html>
<head>
    <title>Website Footer</title>
</head>
<body>
    <h1>Welcome to My Website</h1>
    <footer>
        <p>&copy; 2025 My Website. All rights reserved.</p>
    </footer>
</body>
</html>
```

Project 24: Embedding Google Maps

Objective:

Embed a Google map in your webpage to show a location.

Explanation:

Use the `<iframe>` tag to embed Google Maps into your website.

Source Code:

html

CopyEdit

```
<!DOCTYPE html>
<html>
<head>
  <title>Google Map</title>
</head>
<body>
  <h1>Our Office Location</h1>
  <iframe
src="https://www.google.com/maps/embed?pb=!1m18!1m12!1m3!1d243647.1563
7835863!2d-74.0060!3d40.7128" width="600" height="450"></iframe>
</body>
</html>
```

Project 25: Making a Countdown Timer**Objective:**

Create a countdown timer that shows the remaining time until a specific event.

Explanation:

Use JavaScript to calculate and display a countdown timer.

Source Code:

html

CopyEdit

```
<!DOCTYPE html>
<html>
<head>
  <title>Countdown Timer</title>
</head>
<body>
  <h1>Countdown to New Year</h1>
  <p id="countdown"></p>

  <script>
```

```

        var countdownDate = new Date("Dec 31, 2025
23:59:59").getTime();
        var x = setInterval(function() {
            var now = new Date().getTime();
            var distance = countdownDate - now;
            var days = Math.floor(distance / (1000 * 60 * 60 * 24));
            var hours = Math.floor((distance % (1000 * 60 * 60 * 24))
/ (1000 * 60 * 60));
            var minutes = Math.floor((distance % (1000 * 60 * 60)) /
(1000 * 60));
            var seconds = Math.floor((distance % (1000 * 60)) / 1000);

            document.getElementById("countdown").innerHTML = days + "d
" + hours + "h " + minutes + "m " + seconds + "s ";

            if (distance < 0) {
                clearInterval(x);
                document.getElementById("countdown").innerHTML =
"EXPIRED";
            }
        }, 1000);
    </script>
</body>
</html>

```

Project 26: HTML Form with Drop-down List

Objective:

Create a form with a drop-down list to select options.

Explanation:

Use the `<select>` and `<option>` tags to create a drop-down menu for forms.

Source Code:

```

html
CopyEdit
<!DOCTYPE html>
<html>

```

```
<head>
  <title>Form with Drop-down List</title>
</head>
<body>
  <h1>Choose Your Favorite Car</h1>
  <form action="/submit" method="POST">
    <label for="cars">Choose a car:</label>
    <select name="cars" id="cars">
      <option value="toyota">Toyota</option>
      <option value="ford">Ford</option>
      <option value="honda">Honda</option>
    </select><br><br>
    <input type="submit" value="Submit">
  </form>
</body>
</html>
```

Project 27: Simple Responsive Web Page

Objective:

Create a responsive web page using CSS flexbox for layout.

Explanation:

This project will help you build a flexible, mobile-friendly design using CSS Flexbox.

Source Code:

html

CopyEdit

```
<!DOCTYPE html>
<html>
<head>
  <title>Responsive Layout</title>
  <style>
    .container {
      display: flex;
      flex-wrap: wrap;
      justify-content: space-between;
```

```
    }
    .box {
        flex: 1 1 300px;
        padding: 20px;
        margin: 10px;
        background-color: lightblue;
        text-align: center;
    }
</style>
</head>
<body>
    <h1>Responsive Layout Example</h1>
    <div class="container">
        <div class="box">Box 1</div>
        <div class="box">Box 2</div>
        <div class="box">Box 3</div>
    </div>
</body>
</html>
```

Project 28: Image Hover Effects

Objective:

Create an image with hover effects.

Explanation:

CSS hover effects can be used to change the appearance of an image when the user hovers over it.

Source Code:

```
html
CopyEdit
<!DOCTYPE html>
<html>
<head>
    <title>Image Hover Effect</title>
    <style>
```

```
        .image-container {
            position: relative;
            display: inline-block;
        }
        .image-container img {
            width: 300px;
            height: 200px;
        }
        .image-container:hover img {
            opacity: 0.7;
        }
    </style>
</head>
<body>
    <h1>Hover Over the Image</h1>
    <div class="image-container">
        
    </div>
</body>
</html>
```

Project 29: Creating a Sticky Navigation Bar

Objective:

Create a sticky navigation bar that stays at the top of the page when scrolling.

Explanation:

Use `position: sticky` in CSS to make the navigation bar stick to the top when the page is scrolled.

Source Code:

```
html
CopyEdit
<!DOCTYPE html>
<html>
<head>
    <title>Sticky Navigation Bar</title>
```

```
<style>
  .navbar {
    position: sticky;
    top: 0;
    background-color: #333;
    padding: 14px 20px;
    color: white;
    text-align: center;
  }
  .navbar a {
    color: white;
    padding: 14px 20px;
    text-decoration: none;
    display: inline-block;
  }
  .navbar a:hover {
    background-color: #ddd;
    color: black;
  }
</style>
</head>
<body>
  <div class="navbar">
    <a href="#home">Home</a>
    <a href="#services">Services</a>
    <a href="#contact">Contact</a>
  </div>

  <h2>Scroll Down to See the Sticky Effect</h2>
  <p>Content goes here...</p>
</body>
</html>
```

Project 30: Contact Form with Radio Buttons

Objective:

Create a contact form with radio buttons for selecting options.

Explanation:

Radio buttons allow users to select one option from multiple choices.

Source Code:

```
html
CopyEdit
<!DOCTYPE html>
<html>
<head>
  <title>Contact Form with Radio Buttons</title>
</head>
<body>
  <h1>Contact Us</h1>
  <form action="/submit" method="POST">
    <label for="name">Name:</label><br>
    <input type="text" id="name" name="name"><br><br>

    <label for="gender">Gender:</label><br>
    <input type="radio" id="male" name="gender" value="male">
Male<br>
    <input type="radio" id="female" name="gender" value="female">
Female<br><br>

    <input type="submit" value="Submit">
  </form>
</body>
</html>
```

Project 31: Centering a Div with Flexbox**Objective:**

Center a `<div>` element using Flexbox.

Explanation:

Flexbox allows easy vertical and horizontal centering of elements within a container.

Source Code:

```
html
CopyEdit
<!DOCTYPE html>
<html>
<head>
  <title>Centering a Div</title>
  <style>
    .container {
      display: flex;
      justify-content: center;
      align-items: center;
      height: 100vh;
    }
    .box {
      width: 200px;
      height: 200px;
      background-color: lightblue;
    }
  </style>
</head>
<body>
  <div class="container">
    <div class="box"></div>
  </div>
</body>
</html>
```

Project 32: Floating Elements Using CSS

Objective:

Learn how to use the `float` property to position elements on the page.

Explanation:

The `float` property allows elements to be positioned next to each other.

Source Code:

```
html
```


CopyEdit

```
<!DOCTYPE html>
<html>
<head>
  <title>Floating Elements</title>
  <style>
    .float-left {
      float: left;
      width: 45%;
      margin: 5px;
      background-color: lightblue;
      padding: 10px;
    }
    .float-right {
      float: right;
      width: 45%;
      margin: 5px;
      background-color: lightgreen;
      padding: 10px;
    }
  </style>
</head>
<body>
  <div class="float-left">Left Side</div>
  <div class="float-right">Right Side</div>
</body>
</html>
```

Project 33: Simple Login Page with Background Image

Objective:

Create a login page with a background image.

Explanation:

Use CSS to set a background image and center the login form.

Source Code:

html

CopyEdit

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
  <title>Login Page</title>
```

```
  <style>
```

```
    body {
```

```
      background-image: url('background.jpg');
```

```
      background-size: cover;
```

```
      display: flex;
```

```
      justify-content: center;
```

```
      align-items: center;
```

```
      height: 100vh;
```

```
      margin: 0;
```

```
    }
```

```
    .login-form {
```

```
      background-color: rgba(0, 0, 0, 0.5);
```

```
      padding: 30px;
```

```
      border-radius: 10px;
```

```
      color: white;
```

```
    }
```

```
  </style>
```

```
</head>
```

```
<body>
```

```
  <div class="login-form">
```

```
    <h1>Login</h1>
```

```
    <form action="/login" method="POST">
```

```
      <label for="username">Username:</label><br>
```

```
      <input type="text" id="username" name="username"><br><br>
```

```
      <label for="password">Password:</label><br>
```

```
      <input type="password" id="password"
```

```
name="password"><br><br>
```

```
      <input type="submit" value="Login">
```

```
    </form>
```

```
  </div>
```

```
</body>
```

```
</html>
```

Project 34: Multi-Column Layout with CSS

Objective:

Create a multi-column layout using CSS.

Explanation:

Use the `column-count` property in CSS to display content in multiple columns.

Source Code:

html

CopyEdit

```
<!DOCTYPE html>
<html>
<head>
  <title>Multi-Column Layout</title>
  <style>
    .content {
      column-count: 2;
      column-gap: 20px;
    }
  </style>
</head>
<body>
  <h1>Multi-Column Layout</h1>
  <div class="content">
    <p>This is the first column of content.</p>
    <p>This is the second column of content.</p>
    <p>More content goes here.</p>
    <p>Continue adding text for the multi-column effect.</p>
  </div>
</body>
</html>
```

Project 35: Interactive Quiz with HTML Forms

Objective:

Create a simple interactive quiz using HTML forms.

Explanation:

You'll build a form with multiple-choice questions and a submit button.

Source Code:

html

CopyEdit

```
<!DOCTYPE html>
<html>
<head>
  <title>Quiz</title>
</head>
<body>
  <h1>Test Your Knowledge</h1>
  <form action="/submit" method="POST">
    <p>What is the capital of France?</p>
    <input type="radio" name="question1" value="paris"> Paris<br>
    <input type="radio" name="question1" value="london">
London<br><br>

    <p>Which is the largest ocean?</p>
    <input type="radio" name="question2" value="pacific">
Pacific<br>
    <input type="radio" name="question2" value="atlantic">
Atlantic<br><br>

    <input type="submit" value="Submit">
  </form>
</body>
</html>
```

Project 36: Adding a Background Audio

Objective:

Embed background audio that plays when the page loads.

Explanation:

Use the `<audio>` tag to embed and play audio.

Source Code:

html

CopyEdit

```
<!DOCTYPE html>
<html>
<head>
  <title>Background Music</title>
</head>
<body>
  <h1>Welcome to My Website</h1>
  <audio autoplay loop>
    <source src="background-music.mp3" type="audio/mp3">
    Your browser does not support the audio element.
  </audio>
</body>
</html>
```

Begin your coding journey today visit: [Modern Age Coders](#)